What is claimed is:

1. An instruction scheduling method comprising:

a priority calculation step of calculating a priority of each of a plurality of instructions that are subjected to scheduling, based on dependencies between the plurality of instructions and constraints of hardware resources for processing the plurality of instructions, the dependencies being data dependency, anti-dependency, and output dependency; and

an execution timing decision step of deciding an execution timing of an instruction having a highest priority.

2. The instruction scheduling method of Claim 1,

wherein the priority calculation step includes:

a precedence constraint rank calculation substep of calculating a precedence constraint rank of each of the plurality of instructions, wherein (a) if the instruction has a succeeding instruction which is anti-dependent or output dependent on the instruction, the precedence constraint rank of the instruction is equal to a precedence constraint rank of the succeeding instruction, and (b) if the instruction has a succeeding instruction which is data dependent on the instruction, the precedence constraint rank of the instruction is higher than a precedence constraint rank of the succeeding instruction; and

a resource constraint evaluation substep of judging (i) whether the instruction has a succeeding instruction which is dependent on the instruction, (ii) whether the instruction and the succeeding instruction have an equal precedence constraint rank, and (iii) whether a hardware resource for processing the instruction cannot process the instruction and the succeeding instruction in parallel, and

the priority calculation step raises the precedence constraint rank of the instruction and sets the raised precedence constraint rank as a priority of the instruction if all of the judgments (i), (ii), and (iii) are in the affirmative, and sets the precedence constraint rank of the instruction as the priority of the instruction if any of the judgments (i), (ii), and (iii) is in the negative.

3. The instruction scheduling method of Claim 1,

wherein the priority calculation step includes:

a precedence constraint rank calculation substep of calculating a precedence constraint rank of each of the plurality of instructions, wherein (a) if the instruction has no succeeding instruction which is dependent on the instruction, the precedence constraint rank of the instruction is 1, (b) if the instruction has one or more succeeding instructions which are anti-dependent or output dependent on the instruction, the precedence constraint rank of the instruction is a highest one of precedence constraint ranks of the succeeding instructions, and (c) if the instruction has one or more succeeding instructions which are data dependent on the instruction, the precedence constraint rank of the instruction is a sum of 1 and a highest one of precedence constraint ranks of the succeeding instructions; and

a resource constraint evaluation substep of calculating a resource constraint value of the instruction, by dividing a total number of instructions which are to be processed by a hardware resource for processing the instruction and whose execution timings have not been decided, by a maximum number of instructions that can be processed in parallel by the hardware resource, and

the priority calculation step sets the resource constraint value as a priority of the instruction if the resource constraint value is larger than the precedence constraint rank, and sets the precedence constraint rank as the priority of the instruction if the resource constraint value is no larger than the precedence constraint rank.

4. An instruction scheduling method for sequentially deciding execution timings of instructions that are subjected to scheduling, comprising:

a decision judgment step of judging, after an execution timing of a first instruction is decided, whether an execution timing of a second instruction can be decided so as to be within a predetermined time period, based on a constraint of a hardware resource for processing the second instruction; and

a redecision step of retracting, if the judgment is in the negative, the decision of the execution timing of the first instruction and deciding an execution timing of an instruction other than the first instruction.

5. The instruction scheduling method of Claim 4,

wherein the predetermined time period is expressed by a number of clock cycles,

the decision judgment step includes:

a resource constraint evaluation substep of calculating a resource constraint value of the second instruction, by dividing a total number of instructions which are to be processed by the hardware resource and whose execution timings have not been decided, by a maximum number of instructions that can be processed in parallel by the hardware resource, and the decision judgment step judges in the negative if the resource constraint value is larger than the number of clock cycles.

6. A program conversion method characterized in that:

an input program is converted to an object program including a plurality of instructions, and an execution timing of each of the plurality of instructions in the object program is decided using the instruction scheduling method of one of Claims 1 to 5.

7. An instruction scheduling device comprising:

a priority calculation unit operable to calculate a priority of each of a plurality of instructions that are subjected to scheduling, based on dependencies between the plurality of instructions and constraints of hardware resources for processing the plurality of instructions, the dependencies being data dependency, anti-dependency, and output dependency; and

an execution timing decision unit operable to decide an execution timing of an instruction having a highest priority.

8. An instruction scheduling device for sequentially deciding execution timings of instructions that are

subjected to scheduling, comprising:

a decision judgment unit operable to judge, after an execution timing of a first instruction is decided, whether an execution timing of a second instruction can be decided so as to be within a predetermined time period, based on a constraint of a hardware resource for processing the second instruction; and

a redecision unit operable to retract, if the judgment is in the negative, the decision of the execution timing of the first instruction and decide an execution timing of an instruction other than the first instruction.

9. A computer-executable program for instruction scheduling, having a computer execute:

a priority calculation step of calculating a priority of each of a plurality of instructions that are subjected to scheduling, based on dependencies between the plurality of instructions and constraints of hardware resources for processing the plurality of instructions, the dependencies being data dependency, anti-dependency, and output dependency; and

an execution timing decision step of deciding an execution timing of an instruction having a highest priority.

10. A computer-executable program for sequentially deciding execution timings of instructions that are subjected to scheduling, having a computer execute:

a decision judgment step of judging, after an execution timing of a first instruction is decided, whether an execution timing of a second instruction can be decided so as to be within a predetermined time period, based on a constraint of a hardware resource for processing the second instruction; and

a redecision step of retracting, if the judgment is in the negative, the decision of the execution timing of the first instruction and deciding an execution timing of an instruction other than the first instruction.

11. A computer-readable storage medium storing the program of one of Claims 9 and 10.